Remarks

Applicant appreciates the Examiner making time to quickly fax another copy of Mihata, "Design Rule Verifying System," to the undersigned attorney.

This communication is considered fully responsive to the first Office action mailed November 12, 2003. Claims 1-26 were examined and stand rejected. Claims 1, 12 and 24 were amended. No claims were added. Reexamination and reconsideration of claims 1-26 are respectfully requested.

Claim Rejections – 35 U.S.C. § 103(a)

The Office Action rejected claims 1-3, 5-15, and 17-26 under 35 U.S.C. 103(a) as being purportedly unpatentable over U.S. Patent No. 6,059,842 to Dumarot, et al. ("Dumarot") in view of U.S. Patent No. 5,729,472 to Seiffert, et al. ("Seiffert"). The Applicant respectfully traverses this rejection.

Claim 1 is amended to positively recite "generating a number of rules based on said derived relationships." Dumarot fails to disclose or suggest this recitation. Dumarot discusses adjusting system and application settings based on user-specified quality/performance trade-offs (Col. 7, lines 8-10). Rules are selected from a set of available rules (Col. 7, lines 50-53), such as rules 331, 341, 351 which may be updated based on information provided by application vendors. See e.g., Col. 8, lines 15-18. There is no teaching or suggestion of generating these rules based on derived relationships. Indeed, the Office Action even noted that Dumarot fails to disclose or suggest this recitation.

Instead, the Office Action relied on the teachings in Seiffert. However, Seiffert fails to disclose or suggest the recitations in claim 1 of "electronically deriving relationships over time between monitored system variables and monitored performance of said computer system [and] generating a number of rules based on said derived relationships." Seiffert merely discusses optimizing

Amendment

1.   (currently amended) A method for enhancing performance of a computer system, comprising:

electronically deriving relationships over time between monitored system variables and [[the]] monitored performance of said computer system;

generating a number of rules based on said derived relationships; and

adjusting at least one of said system variables based on said generated number of rules to enhance the performance of said computer system.

2.   (original)   A method as in claim 1, wherein generating said number of rules is based at least in part on a performance goal.

3.   (original)   A method as in claim 1, wherein generating said number of rules is based at least in part on current values of said system variables, and wherein said number of rules recommend incremental changes to said system variables.

4.   (original)   A method as in claim 1, wherein the steps of: (a) deriving said relationships, (b) generating said number of rules, and (c) adjusting said at least one system variable, are iterative.

5.   (original)   A method as in claim 1, further comprising acquiring data for said system variables and the performance of said computer system, wherein said acquired data is used for deriving said relationships.

6.   (original)   A method as in claim 5, wherein acquiring said data comprises:

    a)   gathering said data over time; and

    b)   logging said gathered data, wherein said relationships are derived based on said logged data.

7.   (original)   A method as in claim 6, wherein gathering said data is at discrete points in time.

8.   (original)   A method as in claim 6, wherein gathering said data is in response to an event on said computer system.

9.   (original)   A method as in claim 5, wherein acquiring said data comprises acquiring at least one of the following types of data: configuration data, workload data, and performance metric data.

10.   (original)   A method as in claim 1, further comprising identifying a number of applications on said computer system having variables that affect the performance of said computer system.

11.   (original)   A method as in claim 1, further comprising identifying a number of subsystem components on said computer system having variables that affect the performance of said computer system.   .

12.   (currently amended) A method for enhancing performance of a computer system, comprising:

electronically deriving a plurality of relationships over time between a plurality of monitored system variables and [[the]] monitored performance of said computer system;

generating a plurality of rules based on said plurality of derived relationships; and

adjusting at least one of said system variables based on said generated number of rules to enhance the performance of said computer system.

13.   (original)   A method as in claim 12, wherein the performance of said computer system is based on a plurality of performance metrics.

14.   (original)   An apparatus for enhancing performance of a computer system, comprising:

computer readable storage media;

computer readable program code stored on said computer readable storage media, comprising:

a)   program code for deriving relationships between system variables and the performance of said computer system;

b)   program code for generating a number of rules based on said derived relationships; and

c)   program code for adjusting at least one of said system variables based on said generated number of rules to enhance the performance of said computer system.

15. (original)   An apparatus as in claim 14, wherein said number of rules are generated by said program code based at least in part on a performance goal.

16. (original)   An apparatus as in claim 14, further comprising program code for iteratively deriving relationships between said system variables and the performance of said computer system, and iteratively generating a number of rules based on said derived relationships when an adjustment is made to said at least one system variable.

17. (original)   An apparatus as in claim 14, further comprising program code for acquiring data for said system variables and the performance of said computer system.

18. (original)   An apparatus as in claim 17, wherein at least some of said data is acquired from a configuration file.

19. (original)   An apparatus as in claim 17, wherein at least some of said data is acquired by monitoring said computer system.

20. (original)   An apparatus as in claim 17, wherein said program code for acquiring said data comprises:

  a)   program code for gathering said data over time;

  b)   program code for logging said gathered data, wherein said program code for deriving derives said relationships based on said logged data.

5

21.   (original)   An apparatus as in claim 17, wherein said program code for acquiring said data acquires at least one of the following types of data: configuration data, workload data, and performance metric data.

22.   (original)   An apparatus as in claim 14, further comprising program code for identifying a number of applications on said computer system having variables that affect the performance of said computer system.

23.   (original)   An apparatus as in claim 14, further comprising program code for identifying a number of subsystem components of said computer system having variables that affect the performance of said computer system.

24.   (currently amended) An apparatus for enhancing performance of a computer system, comprising:

        means for electronically deriving relationships over time between monitored system variables and [[the]] monitored performance of said computer system;

        means for generating a number of rules based on said derived relationships; and

        means for adjusting at least one of said system variables based on said generated number of rules to enhance the performance of said computer system.

25.   (original)   An apparatus as in claim 24, further comprising means for acquiring data for said system variables and the performance of said system.

26.   (original)   An apparatus as in claim 25, wherein said acquiring means comprises:

    a)   means for gathering said data over time; and

    b)   means for logging said data, wherein said relationships are derived based on said logged data.

the location at which a rule is evaluated. The rules are defined <u>by a system administrator</u> and are not based on monitored system variables and monitored performance. See e.g., Col. 6, lines 15-16. Although Seiffert discusses several examples of the system administrator defining a rule (see e.g., Col. 9, lines 27-35, Col. 10, lines 25-37, and Col. 11, lines 28-40), none of these examples teach or suggest generating a number of rules <u>based on derived relationships</u>.

For at least the foregoing reasons amended claim 1 is believed to be allowable over the cited references and Applicant respectfully requests withdrawal of the rejection of claim 1.

Claims 2-3 and 5-11 depend from claim 1, which is believed to be allowable. Therefore, claims 2-3 and 5-11 are believed to be allowable for at least the same reasons as claim 1. Withdrawal of the rejection of claims 2-3 and 5-11 is respectfully requested.

Furthermore, claim 2 positively recites "<u>wherein generating said number of rules</u> is based at least in part on a performance goal" and claim 3 positively recites "<u>wherein generating said number of rules</u> is based at least in part on current values of said system variables . . . ." The Office Action cites to Dumarot to reject claims 2 and 3. However, the Office Action previously noted in the rejection of claim 1 that Dumarot does not teach "generating a number of rules." Accordingly, the rejection of claims 2 and 3 only in view of Dumarot is improper.

Claim 12 is amended to positively recite "<u>electronically deriving a plurality of relationships over time</u> between a plurality of <u>monitored system variables and monitored performance</u> of said computer system [and] generating a plurality of rules <u>based on said plurality of derived relationships</u>." As discussed above with regard to claim 1, the cited references taken alone or in combination fail to disclose or suggest at least these recitations. Therefore,

amended claim 12 is believed to be allowable over the cited references and withdrawal of the rejection is respectfully requested.

Claim 13 depends from claim 12, which is believed to be allowable. Therefore, claim 13 is believed to be allowable for at least the same reasons as claim 12. Withdrawal of the rejection of claim 13 is respectfully requested.

Claim 14 positively recites "program code for generating a number of rules based on said derived relationships." Dumarot and Seiffert fail to disclose or suggest this recitation. Dumarot merely discusses a company or system administrator managing and updating rules based on information provided by application vendors. See e.g., Col. 8, lines 15-18. Seiffert merely discusses rules which are defined by a system administrator. See e.g., Col. 6, line 30-31, 37. Neither of these references discloses program code for generating a number of rules based on derived relationships. Therefore, claim 14 is not properly rejected as being unpatentable over the cited references for at least this reason and withdrawal of the rejection is respectfully requested.

Claims 15 and 17-23 depend from claim 14, which is believed to be allowable. Therefore, claims 15 and 17-23 are believed to be allowable for at least the same reasons as claim 14. Withdrawal of the rejection of claims 15 and 17-23 is respectfully requested.

Claim 24 is amended to positively recite "means for electronically deriving relationships over time between monitored system variables and monitored performance of said computer system [and] means for generating a number of rules based on said derived relationships." As discussed above with regard to claim 1, the cited references taken alone or in combination fail to disclose or suggest at least these recitations. Therefore, amended claim 24 is believed to be allowable over the cited references and withdrawal of the rejection is respectfully requested.

Claims 25-26 depend from claim 24, which is believed to be allowable. Therefore, claims 25-26 are believed to be allowable for at least the same reasons as claim 24. Withdrawal of the rejection of claims 25-26 is respectfully requested.

The Office Action rejected claims 4 and 16 under 35 U.S.C. 103(a) as purportedly unpatentable over U.S. Patent No. 6,059,842 to Dumarot, et al. ("Dumarot") in view of U.S. Patent No. 5,729,472 to Seiffert, et al. ("Seiffert") and further in view of the English Abstract of JP403010379, cited as Mihata, "Design Rule Verifying System ("Mihata"). The Applicant respectfully traverses this rejection.

Claim 4 depends from claim 1 and claim 16 depends from claim 14. As discussed above, claims 1 and 14 are believed to be allowable over the cited references. Therefore, it follows that claim 4 and claim 16 are also allowable for at least the same reasons as the independent claims.

Furthermore, claim 4 positively recites "(a) deriving said relationships, (b) generating said number of rules, and (c) adjusting said at least one system variable, are iterative." Claim 16 positively recites "program code for iteratively deriving relationships between said system variables and the performance of said computer system, and iteratively generating a number of rules based on said derived relationships when an adjustment is made to said at least one system variable." Mihata merely states that "such an inconvenience as a vefification [sic] processing by a computer and a correcting work to the contradictory design rule are repeated extending over a large number of times due to being unfamiliar with the contradictory contents can be evaded, and the efficiency of the correcting work can be improved." However, Mihata does not teach or suggest the recitations in claims 4 or the recitations in claim 16, much less the iterative nature of these recitations.

Conclusion

The Applicant respectfully requests that a timely Notice of Allowance be issued in this matter.

Respectfully Submitted,

Dated: 2-11-04   By: Mark D. Trenner

Mark D. Trenner
Reg. No. 43,961
(509) 324-9256